LIBRARY
UNIVERSITY MICROFILMS, INC.

Programed Instruction

Volume 1 (September 1961 - June 1962)

the bi-monthly bulletin of

THE CENTER FOR PROGRAMED INSTRUCTION, INC.
365 West End Avenue at 77th Street
New York 24, New York

Available for $1.00 a year if prepaid
or $2.00 a year if invoice required.

Programed Instruction
ISSUED BY THE CENTER FOR PROGRAMED INSTRUCTION

May 1961, Volume 1
Bulletin Number 1

Out of the Laboratory, into the Classroom:

The school year 1960-61 can be marked down as the year when programed instruction got out of the psychological laboratories and into the classrooms. In school systems all across the nation, children and teachers received their first exposure to an embryonic but immensely promising teaching technique.

Exposure was not limited to teachers and children. Newspapers and magazines brought word of the new technique with news stories and advertisements aimed at those not immediately concerned with education. More or less informed discussions of "teaching machines" were to be heard not only in university psychology departments, but also in professional gatherings, at PTA's, and over dinner tables. A Wall Street market letter recommended the purchase of shares in companies which were producing programs, and estimated, presumably on some evidence, that one of the program publishers would gross 2½ million dollars from the sale of programs alone in 1962.

As an aid to research, the National Education Association published a compendium of articles on the subject — *Teaching Machines and Programmed Learning,* edited by Lumsdaine and Glaser — and saw the 700-page, highly technical book go through three printings in less than a year. Material is presently being collected for a supplement to this "bible" of the field. At the Center for Programed Instruction, the mail in early spring 1961 increased at a phenomenal rate as educators sought information about programing. During the past six months, members of the Center have reported officially at twelve national conferences, seventeen regional meetings, and twenty-eight large-scale parent-teacher gatherings. Many similar invitations had to be turned down.

By the end of the academic year, several dozen school systems will have experimented with programed instruction. High-school students in Denver, for example, have been working with the *English 2600* program and the great masses of data accumulated during the year's work will be assembled and evaluated over the summer. Toward the end of the school year, Denver will give the mathematics program *Sets, Relations and Functions* to some seven hundred seventh-graders in a controlled study organized with the help of the Center for Programed Instruction. Meanwhile, teachers within the Denver system are writing programs to teach English grammar, and developing a unit on the Constitution.

In the Roanoke, Virginia schools, a study is being conducted to determine the effectiveness of programs in Algebra, Geometry and Trigonometry prepared at Hollins College. Similar studies, with different experimental designs, are being conducted under sponsorship of the Hollins group elsewhere in the nation. Preliminary reports have been distributed. More detailed research findings are awaited.

Experiments are under way to determine the extent to which programers must allow for individual differences in intelligence in constructing their programs. One such study, in the schools of Concord, Mass., produced differences in the achievement of matched groups of students who worked along different "tracks" of a vocabulary program.

Other studies attempted to explore the usefulness of programing in solving stubborn educational problems at the opposite end of the intelligence scale. The Devereaux schools created programs for both mentally retarded and exceptionally talented children. In New York City, the junior high school division tried out a reading program for seventh-graders whose test scores placed them on a third-grade level.

The list of subjects programed has grown rapidly, and now includes phys-

Continued on page 6

Inside Opinion

QUESTION: Does the utilization of teaching machines and/or programs necessarily commit a school to a particular philosophy of education?

ANSWER: All philosophies of education are directed toward efficient learning. No matter what the local program, teaching machines or programed instruction without machines can be used effectively for part of the task of education. They can be used in connection with any other type of instruction. There is no evidence that the student is made dependent on machines or programs. Indeed, the available evidence indicates quite the contrary. Students who have developed confidence and competence through programed instruction make excellent students in discussions and other types of instructional arrangements. The processes utilized by teaching machines are present whenever the student studies. He looks at a visual presentation or listens to an auditory one, he makes tentative responses, refers again to the material for additional help if needed, eventually emits a response and seeks some kind of confirmation. These are age-old processes; the machines simply permit the student to engage in them more efficiently.

B. F. Skinner

Edgar Pierce Professor of Psychology, Harvard University, author of **Science and Human Behavior,** "father" of the current interest in programed instruction.

ANSWER: It depends, of course, on how you interpret the expression 'philosophy of education.' Suppose one holds that the only worthwhile learning is that which comes from a student's efforts to deal with his own immediate problems, i.e. those problems that arise outside his school experience. Suppose this same person holds that every student's problems are unique to him. If one held that sort of philosophy of education, he would find little use for teaching machines or programs. Imagine, at the other extreme, a teacher who holds that worthwhile learning is always painful, that without suffering there is no real teaching or learning: such a person might find teaching machines far too humane for his philosophy of education.

Continued on page 2

The Center for Programed Instruction, Inc. hereby grants to all non-profit organizations, including schools with a regular curriculum, the right to reproduce any part of this work. Reproduction by any other person, individual, corporate or otherwise, of any part of this work, without the permission of the Center is strictly prohibited.

INSIDE OPINION — (Cont. from page 1)

In both these hypothetical instances, we have interpreted 'philosophy of education' to mean a kind of irrational dogmatism, and it is clear that people can have dogmatic objections to the kind of instruction given by teaching machines. If, however, one means by 'philosophy of education' a reasoned system of basic beliefs about the purpose and content of education, then surely teaching machines are perfectly neutral. They can be used for good educational purposes or poor. One can develop programs for teaching important concepts or trivialities.

Personally, I think the connection between deliberate programing and philosophy of education is rather the reverse of that implied in the original question. Programing gives us control over the content and processes of learning, it therefore requires us to think more deeply and cogently than ever before on basic educational questions: What are we trying to teach? Is it really worth teaching? To some? To all? What activities other than pure instruction should we include in a total school program? Etc. Programing doesn't commit us to a philosophy of education; it forces us to build one.

James McClellan

Associate Professor of Education, Teachers College, Columbia University, Philosopher, participant in a symposium "Automated Teaching — A Philosophical Approach".

QUESTION: Should programers attempt to construct program variations to provide for individual differences among students?

ANSWER: Linear programing assumes that all students beginning a particular program have equivalent backgrounds and experiences. It is assumed that all students know the words that are used, have a given set of facts already in their repertoire, and have a familiarity with a selected number of principles. This assumption is made in education generally: that students have prerequisite skills before entering a course. For example, it is assumed that they can solve simple problems in algebra before studying physics.

In order to accomodate students of different backgrounds and skills, it is necessary to bring all students up to the same levels before starting on the material which is new for everyone. This can be achieved in several ways, by branching, by having pre-programs (such as the review unit on mathematics which we furnished with our physics program) or by selecting students for a given program on the basis of an achievement test. Alternately, it is possible to supply students representing a range of abilities with the same program at various points in their academic careers. A well prepared fourth grader, for example, might be well equipped to begin physics at the same time as another student in the twelfth grade who has had difficulty in learning.

David J. Klaus

Researcher, American Institute for Research, author of articles on the use of self-tutoring materials in high school science courses.

ANSWER: Although there is no experimental evidence available at the present time upon which to base an answer to this question, nevertheless, since I believe that this is the question that the Newsletter would like to have answered, I shall hazard an opinion. This I shall do with the full knowledge that there may be no single answer to the question, that the answer might well depend on the kind of subject matter programed, the age of the students using the program, as well as the method of programing employed.

Until there is evidence which indicates otherwise, I suggest that the programer forget about the problem of individual differences while he is in the process of developing his program. He should aim his program at the "average" or slightly below average student and develop only one form of the program. Later, when the program is tested, he may be pleasantly surprised to learn that the same program is versatile enough to be used satisfactorily not only with average students, for whom it was prepared, but also with slow and bright students, the slow students taking more time and the bright students, less time, to complete the program than the average students. Individual differences may turn out to be only a very minor concern for programers of self-instructional courses.

Irving J. Saltzman

Associate Professor of Psychology, University of Indiana, principal investigator in the Office of Education project to develop and evaluate a self-instructional program for teaching Russian to high school and college students.

Programed Instruction is published by The Center for Programed Instruction, Inc., 365 West End Avenue, New York 24, N. Y.

May 1961 Volume I

Bulletin No. 1

Robert T. Filep
Editor

The Cost of publishing this Bulletin was paid in part under a grant from the Carnegie Corporation of New York.

Printed in U.S.A.

Responses

Dear Mr. Komoski,

Last spring the administrators of our district heard Dr. Galanter* tell about programing for teaching machines. No meeting I have ever attended has given me more ideas to put to work in the classroom than that one.

He emphasized the importance of programing, whether used with elaborate machines, simple machines, commercial materials, or teacher-made materials.

I got five rather simple ideas from that meeting which make sense, and I try to keep them before me while I plan lessons, make worksheets, and teach:

1. Break down the learning into tiny steps leading from what they know to what you want them to know.
2. Have them *do* something.
3. Don't let them make mistakes — by keeping the steps tiny and if necessary by clues.
4. If they do make a mistake, have *them* correct it immediately.
5. If they get it right, reinforce immediately — by letting them know they got it right and giving them another problem.

They made sense out of some of the most successful techniques I had seen and used, and they pointed the way to more successful teaching.

One more thing that occurs to me as I look back on a few months of teaching with the idea of step-by-step programing in my head, is the fact that this programing is an art, a difficult one. I look back at worksheets I have made, and examine the responses the kids gave, and I say, "Whoops, I took too big a jump there," or "I shouldn't have left out that word," and I realize that the programing not only helps the kids learn, but it helps me learn about *how* the kids learn.

Sincerely,
Dr. Sanger B. Steel

(4th grade teacher, former elementary school principal, Niskayuna, N. Y.)

* Eugene Galanter, editor, *Automatic Teaching: The State of the Art.*

Programing for Your Classroom

"I have some real reservations as to the over-all profit to the student and to education in instances in which a teacher, who has never constructed a program and who does not understand the process involved in them, uses programs for his students that someone else has prepared for him. I do not necessarily mean that the resulting situation will make things worse off than they are now, but it certainly will not result in the exploitation of the truly great possibilities opened as a result of the development of programing techniques."

JOHN A. BARLOW

We have been cursed by all this talk about machinery. Machines, unless you know them well, are rather awesome and fearful things. The words "teaching machine" call up a terrifying image in the minds of people who spend their lives helping children and adolescents who are trying to learn. So let us forget about the machines for a while. There is always going to be a place for mechanical aids in the classroom – for film projectors, tape recorders, galvanometers and the like. The teaching machine will be one of these – an occasional help, in some subjects, for people who know how to use it. Programing foreign languages, for example, or programing for pre-readers, requires a "teaching machine." But the machine doesn't teach; it can't. Even where a machine is used, the program does the teaching. Work done, for instance, at Bell Labs and at the Center for Programed Instruction has demonstrated clearly that for most subjects the program printed as a pamphlet or a book is just as effective as the program prepared for a "machine." In both cases, of course, a human being writes the program. Once the machinery has been removed, programing is easier to understand for what it is: a way of teaching.

In a sense, then, programed instruction is the systematization of an excellent teaching procedure. But it is not a gimmick, or a superficial technique for applying something teachers have known for a long time. In many ways, programing is fundamentally new. It originated, not in classrooms, but in the laboratories of experimental psychologists, and its foundations lie deep in the development of learning theory.

The Center for Programed Instruction was formed by teachers – not by psychologists, though there are experimental psychologists on our staff – in the belief that the time had come to carry programs out of the laboratory and into the classroom. Its central purpose is to gather and disseminate information about what is happening in this very new field, and to train teachers and psychologists who wish to program material for classrooms. Among its first tenets is the belief that programs can be used most successfully by teachers who have some insight into programing. The Center has urged from the beginning that every teacher try to learn something from a program – something new to him, in an unfamiliar field – before using a program with students. Doing is still a good way of learning. A brief program suitable for such double-purpose self-instruction is CPI's *A Programed Primer on Programing.*

Constructing A Program

But the good teacher is not limited to the consumption of extant programs as a way to learn about them; you can also do-it-yourself. A day's classroom lesson is a kind of program. If you present part of your lesson plan to your students in the form of a written program, you will learn about programing in a most effective way. You may even learn something about teaching; we did.

Most teachers come to a lesson with a clear idea of the questions they will wish their students to be able to answer when the lesson is over. These are the key questions, the ones which will test whether or not the lesson came across, whether or not the students learned what was taught. Sometimes they reach beyond this objective, and seek to find out whether the students can *use* in another context what they have just discovered. But the first purpose of questioning the class is usually verification: have they learned it, do they understand it, what more do I have to teach? Programing is a way to get correct answers to this sort of question from (hopefully) every student in the class. The program, like the lesson, leads up to these confirmatory questions. And, like the teacher, the program leads the student to answer these questions correctly: it is not a test, but a part of the process of instruction.

Lessons are most often presented by textbooks or by the teacher, speaking to the class. In both cases, it is hoped that the student *responds* – absorbs what he reads, thinks about what he hears. Programing is a way to guarantee a response. The student must fill in a missing word or answer a question before he can move on to the next item or "frame". If this paragraph were part of a program, we should say, "he must————." And you would have to fill in the word "respond" before you moved on to the next paragraph.

The technique of programing (at the present time it is more than a technique; it is actually an art) is to ask a student a number of such questions, phrased and graded in such a way that he gets almost all of them right. The student knows he is right when he checks his answer with the correct answer, given him immediately, by the push of the button on a machine, or on the back of the question card, or on the next page of a programed text. Psychologists are convinced that when a student's right answer is immediately confirmed, he will be more likely to make that right answer again the next time the question comes around. You are able to help the student also by immediately correcting a wrong answer (which you can't do *immediately* in a test or a homework assignment) – but you help him a great deal more if he finds the right answer at the first crack.

Continued on page 4

PROGRAMING FOR YOUR CLASSROOM
(Continued from page 3)

How to Begin

Restrict yourself to a very small area of subject matter; for instance, agreement between verbs and such pronouns as 'each', 'either', and 'nobody'. Go over your notes, textbooks, and the assignments you use in teaching this material and determine just what you want to teach. Write down in a sentence each point you wish to make, making sure that there are *no gaps* – that each point develops from the one before. Then drop a key word or phrase from each sentence, leaving blanks which the student must fill in. These sentences represent what you expect the student to know when he masters the material. Now your job is to get him there. You may sometimes need twice as many examples for practicing as you ordinarily assign for this subject. Sometimes you have to break what looks like a single point into half a dozen sentences, because the subject is actually trickier than you thought.

How will you teach him the material in this unit? One consideration, of course, is the grade level of the student. Then, you may have several approaches to teaching subject matter; your techniques and personality will manifest themselves in the program. There are actually many different ways of programing a unit, and in future issues of this Bulletin we will discuss some of these methods at length.

Programing may be frightening at first; it leaves a teacher no easy escape. If the student answers the question wrong, something must be wrong with the program. By definition, the question which pulls wrong answers from any significant proportion of the students must be a bad question. Programing is hard on a teacher's ego.

If at First . . . . .

But a teacher *knows* when something has gone wrong with a program, and knows where it has gone wrong, as he can almost never *know* precisely what has gone wrong when a lesson is presented in the conventional way. Because you have a steady written record of how each student is responding individually, you know where, and may be able to find out why, a given piece of teaching has not succeeded. Nobody, however brilliant or highly trained, gets a program right the first time – or the second time. Programing, like all good teaching, requires great patience. There are no perfect programs yet. Even the published programs will go through a lot of revision in years to come, as their authors readily admit. Programing, despite some publicity to the contrary, is not a fully matured technology; it is still growing.

On the next page you will notice the heading FAULTY FRAMES, and below it a sequence of items from a program written by a trained and experienced programer – a sequence that didn't work. Such a column will be a permanent feature of this bulletin, partly to soothe the anguished breasts of programers, partly in hopes that the accumulation of such detailed data will help people induce general rules as to what is trustworthy and what is dubious in an item.

If you decide to try a little programing on your own, let the Center know how you did with it – what sorts of problems you encountered, and how you solved them (if you solved them). We'll be glad to give you what advice we can, and to look over your program – if you provide us with data about specific problems you encountered in your trial runs. If you're particularly interested in programing, and wish detailed guidance, you might attend one of the summer workshops in programing to be given at various colleges and universities this Summer.

Committee Sets Initial Criteria for Programed Materials

The following is a statement on self-instructional materials and devices released by a joint committee of the American Educational Research Association, the American Psychological Association, and the Department of Audio-Visual Instruction of the N.E.A. at the recent D.A.V.I. Conference held in Miami, Florida.

The use of self-instructional programed learning materials in teaching machines and similar devices represents a potential contribution of great importance to American education. But this contribution can best be realized only if users have information with which to evaluate self-instructional materials. Accordingly, the following interim guide-lines have been prepared.

1. **Teaching machines do not, in themselves, teach. Rather, the teaching is done by a program of instructional materials presented by the teaching machine. Any evaluation of a teaching machine thus requires an assessment of the availability and quality of programs for each type of machine, as well as its mechanical dependability.**
2. **A variety of programed materials is becoming available, but not all programs will fit all machines. Thus only those programs compatible with a particular machine can be considered as available for use with it. A list of commercially available programs and devices can be obtained by sending a request to the Department of Audio-Visual Instruction, National Education Association.**
3. **In evaluating the specific content which a self-instructional program purports to teach, the program can be examined to determine what the student is required to do and whether the student's responses reflect the kind of competence which the educator wishes to achieve. Like other educational materials, programs labeled with the name of a particular subject matter vary widely with respect to content and instructional objectives.**
4. **Just any set of question and answer material does not constitute a self-instructional program. One major type of self-instructional material proceeds by small steps requiring frequent student responses. These steps can be examined to see if they embody a careful, logical progression of the subject matter. Items in such a program are designed so that the student will respond to the critical aspects of each item or will perform the important operation which that item was meant to teach. Furthermore such programs generally provide a wide range of examples illustrating each principle or concept.**
5. **Self-instructional materials are designed to adapt to individual differences by allowing each student to proceed at his own rate. Some types of self-instructional materials further adapt by "branching" to alternate materials. For this purpose, questions are designed to diagnose the student's needs, and to provide alternate material suited to these needs. The material is designed so that the choice of answer to a particular question determines which items will be presented next. Incorrect answers take the student to items containing information designed to correct the error before continuing through the sequence.**

(Continued on page 7)

BAD SEQUENCE

29. An anti-Communist is a person who works against the ideas of the Communists. The prefix _____ means "against".

* * * * *

31. A person who hated (was "against") the Nazi party in Germany was an _____-Nazi.
32. A kind of germ-killer that works "against germs" contained in dirt or filth is called an _____septic.

* * * * *

34. *Anti* words are opposites: *Nazi* and *anti-Nazi*, *septic* and *antiseptic*. An opposite place that isn't "against" anything is the place at the opposite end of the globe from the Arctic which we call the _____arctic.
35. A strong feeling *against* someone (the opposite of *sympathy*) is _____pathy.

25% of the students said *antisympathy*

CORRECTED SEQUENCE

34. *Anti* words are opposites; *Nazi* and *anti-Nazi*, *septic* and *antiseptic*. An opposite place that isn't "against" anything is the place at the opposite end of the globe from the Arctic which we call the _____arctic.

34a. *Antonym* and *synonym* are opposites. The opposite of *antonym*, meaning "opposite in meaning", is *synonym* meaning "_____ in meaning".

34b. Actually no two words have *exactly* the same meaning; there is always some difference between them. Although *syn* means "alike", it is more correct to say "similar". We should say that *synonyms* have "_____" meanings.

34c. Listening to some people's troubles can arouse very strong feelings in you. You share their discouragements and fears. You have "similar feelings", or you feel _____pathy for them. (The prefix *syn* ends with *N* and the root begins with *P*.)

35. A strong feeling *against* someone (the opposite of *sympathy*) is _____pathy.

Faulty Frames

This sequence, part of an 8th grade vocabulary program, moves from familiar to unfamiliar words. The final item "tests" the efficacy of the sequence, because few 8th graders will know the word 'antipathy'.

Errors on item 35 fell into two groups: 25% of superior readers said "antisympathy", while a smaller group responded "sympathy".

At first glance, the item seemed suited for the wastebasket. After analysis, however, the programer decided not to discard it, for the following reasons:

1) The item is not in the least ambiguous or unclear. It is a legitimate extension of the content of the lesson – the prefix *anti-*. There is no reason why a programer should not be able to extend the mastery of *anti-* to the free derivation of the new word antipathy. *The programer felt she should not revise the educational goal downward toward what could be easily achieved, but should revise the techniques upward to achieve the goal.*

2) The smaller group of students saying "The opposite of sympathy is sympathy" presented a separate problem. Examination of their answers to other questions of this type showed that they often made such mistakes. They lacked skills which the programer had assumed and would have to be taught these skills.

3) *The students who said "antisympathy" had not, on close examination of the items, made a mistake. They had done exactly what they had just been taught to do.* Look at the logic of item 34, and extend it to item 35. The students had operated logically; *the programer had blundered in analyzing the subject matter.*

The insertions which cured the bad sequence are given above. The prefix *anti-* is presented with its opposite, *syn-*, on the same root (34a) and the word 'sympathy' is shown to be an analyzable compound (34c). Meanwhile, the programer constructed a new sequence, too long to be shown here, to teach the skills which would reduce the probability of the answer "sympathy" in frame 35. The new sequences worked. Presented to a heterogeneous group, including average and below-average students, the revised program produced the response "antisympathy" in only 6% of the cases.

Dr. Susan Meyer Markle, programer; and editor

The goals could have been revised downward for this sequence by giving students the word 'antipathy' earlier in the program, thus reducing its novelty and increasing the likelihood of a correct response. But the programer wanted the students to learn concepts, not just vocabulary items.

Students' behavior teaches the programer. We learn from error rates those areas where we teach too fast. We learn from analysis of the *content* (not the *percent*) of errors those areas where our teaching is faulty.

If you find that one of your frames produces a high error rate, you should ask yourself the following:

Continued on page 8

OUT OF THE LABORATORY
(Continued from page 1)

ics, chemistry, German, Spanish, Latin, Russian, creative writing, art appreciation, and many others, in addition to those already mentioned.

In Western Pennsylvania, the American Institute for Research is conducting an experiment to determine whether programs can be used to stimulate independent thinking and problem-solving. The students involved in the experiment are in the junior-high age.

Programs are being tried out in nursery schools and in colleges. At the Dalton School in New York, four- and five-year-olds, using a simple teaching machine, have learned to manipulate number concepts from 1 to 9, without the use of written numerals. In the colleges, students are testing the effectiveness of programs to teach Russian, French, Chinese, Hebrew, freshman English, logic, biology, kinetics, statistics, philosophy, music and psychology. On the west coast, a number of programs have been designed and used to teach workers new skills.

More than one hundred companies are making teaching machines of one description or another, and are in the process of testing their devices. The "recall" machines are being used to present spelling programs in elementary schools. In South Huntington, Long Island, another spelling program will be tried out with and without machines, in class and at home, which will help to determine the value of programed homework.

The list of current experiments in the field could be extended to much greater length, and still be incomplete. Indeed, though the Center for Programed Instruction attempts to maintain contact with all groups working in this field, it is likely that there are a number of studies which have not come to the attention of the Center. Though this bulletin accepts a first obligation to keep the teaching profession in touch with programing, it will in addition serve as a means of keeping programers in touch with each other. The Center will appreciate reports from readers on experimental work in programing and will publish in this Bulletin as much as possible of the news and opinion they contain.

RECENT RESEARCH

Roe, A. et al, "Automated Teaching Methods Using Linear Programs." Automated Learning Research Project, Department of Engineering, University of California, Los Angeles, California. December, 1960.

This study compared the effects of (a) teaching machines, programed textbooks, programed lectures, and a standard lecturer, (b) free response vs. multiple choice responses, and (c) individual booth vs. classroom.

Subject Matter:

Mathematics. The program used was a 192 frame sequence on elementary probability. A pilot study involving 51 freshmen engineering students was conducted to check the comprehensibility, reliability, and validity of the programed materials.

Subjects:

186 freshmen engineering students at the University of California at Los Angeles. Subjects had previously taken an engineering aptitude test which was used to divide them into four ability levels.

Procedures:

The modes of presentation were: (a) two types of teaching machines — a Skinner-type constructed response and a multiple-choice response; (b) two types of programed textbooks — one requiring a response and supplying reinforcement in the form of the correct answer on the next page; the other presenting the items as in a conventional text and requiring no overt response; (c) two types of lecturers — lecturers who followed the sequence of the program (Programed Lecturers) and a conventional lecturer.

A 20-item test (free response items) was given immediately upon completion of the program or lectures.

Results:

(1) No significant test differences were found between the students using the constructed-response and multiple-choice machines and the programed textbook with responses.

(2) No significant differences were found between machine and programed textbooks.

(3) All program groups performed significantly better than the group that received conventional lecture.

(4) The students in the lecture groups took less time than the mean time required by the machine and programed textbook groups.

(5) Multiple-choice and programed textbook groups who did not have to construct responses needed less time than students who constructed answers in machine and programed textbook groups.

(6) While the students in the lower aptitude quartiles did not score as high as the students in the upper quarters, there was no indication that any one of the methods was particularly better for a particular aptitude level, on the basis of either the criterion variable or learning time.

(7) Student attitude towards the various teaching methods correlated neither with aptitude nor with performance on the criterion test.

Comments:

This report could serve as a model of experimental design and reporting in this field. It includes the raw data, the instruments applied to it, pictures of the encironments, and the results of all statistical computation.

Assessing the relative value of overt response with reinforcement as against reading without writing the response, the authors stated: "overt student response did not enhance student learning but merely increased the time necessary for performing the learning tasks." They add, however, that "it would be imprudent to generalize from this series of experiments."

Replication of this study with longer programs should be among the first items of business for psychologists interested in this field. Such studies should also, ideally, compare the retention and (if possible) the transfer of the material learned by students who did make overt responses as against students who did not.

Continued on page 7

Eigen, L. D. and Komoski, P. K., "Research Summary #1 of the Collegiate School Automated Teaching Project." New York, N. Y. June, 1960.

This study investigated the relative effects of teaching machines as against programed texts, and the influence of grade level on achievement in learning new material.

Subject Matter:

Mathematics. The program was **Sets, Relations and Functions,** a 707-frame sequence by Lewis D. Eigen.

Subjects:

74 high-school students (25 in 9th, 23 in 10th, and 26 in 11th grade). The mean Otis IQ for the 74 was 126.8.

Procedures:

Machine and text groups were matched on IQ by grade level. Material was taught by program with no supplementary instruction provided. Tests were given immediately on conclusion of the program. One test covered materials specifically taught; the other covered materials not specifically taught but requiring application of the concepts taught by the program.

Results:

(1) In the comparison of machine as against programed text, results were similar to those of Roe et al., above. The slightly greater time required by students who worked on the machines was attributable to mechanical difficulties with the specific machines.

(2) All grade levels scored about equally well in the test on the material specifically taught in the program, but the older students scored significantly higher on the test involving transfer of learned concepts to new situations.

Comment:

This study casts grave doubt on the usual contention that particular materials should be taught at one grade level rather than another. The data indicated little effect of grade level on time needed to complete the program. Student time to complete the program ranged from 2½ to 7 hours. This study opens the entire question of grade-placement of material to research by programing.

The report fails to give intercorrelation data on the relations between IQ and other variables.

Hughes, J. H., "The Effectiveness of Programed Instruction: Experimental Findings for 7070 Training." International Business Machines Corporation. January, 1961.

This study compared the effectiveness of programed and conventional classroom instruction.

Subject Matter:

The first 15 hours of a 7070 training course for customer engineers were presented (1) by the conventional IBM classroom method (lecture-discussion) and (2) on a 719 Frame programed textbook.

Subjects:

112 men taking a company course in customer engineering.

Procedure:

Two classes of 21 students each were designated "control classes" and put through the company's usual lecture-and-discussion courses. In each of the three subsequent months, pairs of classes undertook to learn the same material from a 719-frame programed text. Instructors in the experimental classrooms confined their activities to distributing the materials and answering a few questions. All classes took a comprehensive two-hour examination (mostly free-response questions) at the conclusion of the course, and filled out an attitude questionnaire.

Results:

(1) Students who worked with programed texts required 11 hours to complete their work, although it is not reported if this is an average time. The lectures and discussions took 15 hours.

(2) Students who worked with programs achieved a mean score of 95.1 on the test, while students in the conventional classes had a mean score of 86.2. This difference was found to be statistically significant at the .01 level.

(3) The attitude questionnaire, filled out anonymously, showed favorable attitudes toward programed instruction on the part of those who had worked with programs.

Comment:

Several variables which might have affected the experimental results were not controlled. The experimental and control groups were not run at the same time. In addition, the experimental group was superior to the control group on several variables which were positively correlated with performance on the achievement test e.g. educational level, aptitude and previous computer experience. Only educational level was controlled statistically.

The results, though encouraging, cannot be taken as conclusive evidence for the superiority of programed instruction.

One of the characteristics of the studies reported is that they all included bright high school students, college students, or adults. There is an obvious need for replication of these experiments with younger children and youngsters of lesser ability.

Reprints of the IBM and Collegiate studies may be secured by writing to the Center. For further information about the other study reported, we suggest you contact the authors.

CRITERIA REPORT
(Continued from page 4)

6. An important feature of almost all self-instructional materials is that a record of the student's responses provide a basis for revising the program. The prospective purchaser should ask about the extent to which revision has been based on student response and how the preliminary tryout was conducted.
7. The effectiveness of a self-instructional program can be assessed by finding out what students actually learn and remember from the program. The prospective purchaser should find out whether such data are available and for what kinds of students and under what conditions the data were obtained.
8. Active experimentation with self-instructional materials and devices in school systems is to be encouraged prior to large scale adoption.

The statement has been approved for release by the D.A.V.I. and A.E.R.A. groups. It is pending final approval for release by the Board of Directors of the A.P.A., which will hold its monthly meeting after this issue has gone to press.

This statement was prepared by the following committees, under the joint chairmanship of A. A. Lumsdaine:

American Educational Research Association: Robert Glaser, Evan R. Keislar, A. A. Lumsdaine, and Harry F. Silberman.

American Psychological Association: Richard S. Crutchfield and James G. Holland.

Department of Audio-Visual Instruction, National Education Association: Jack V. Edling, Edward B. Fry, and Wesley C. Meierhenry.

Summer Workshops

During the summer, a number of colleges and universities will offer workshops on programing. Among them are Harvard, the Universities of Denver, Indiana, Pittsburgh, Utah, Oregon College of Education, and San Francisco State College. Six such workshops will be conducted in collaboration with the Center for Programed Instruction. They will be held at the following institutions:

June 19th-30th: University of South Florida, Tampa, Florida (Att: Dean J. A. Battle)

July 3rd-14th: Rhode Island College of Education, Providence, R. I. (Att: Edward F. Wilcox, Asst. Commissioner of Education, Roger Williams Building, Hayes Street, Providence, R. I.)

July 17th-August 4th: Graduate School of Education, University of Chicago, Chicago 37, Ill. (Att: Dr. Francis Chase)

July 31st-August 11th:
Mills College of Education, 66 Fifth Avenue, New York, N. Y. (Att: Miss Muriel Poland)
Teachers College, Columbia University, 525 West 120th Street, New York 27, N. Y. (att: Dr. Max Brunstetter)

August 14th-25th: The Harley School, 1981 Clover Street, Rochester 18, N. Y. (Att: Mr. Rowland B. McKinley, Jr.)

The workshop in Rhode Island is restricted to a selected group from that state; the others are open to all qualified teachers, administrators, and students. Tuition fees will vary somewhat, according to the regulations of the host institution. All applications should be sent directly to the institution the applicant wishes to attend.

The workshops will be a permanent feature of the Center's calendar, at least for the foreseeable future. At present, there are few courses in programing available as part of any university or teachers college regular catalogue. More courses will probably be introduced within a few years, but even then there will be a need for summer workshops to help teachers who are unable to take such courses during the academic year. The experience of the various reform movements in mathematics and science indicates that the summer workshop is far and away the most efficient way to prepare the introduction of new ideas and procedures into the classroom.

Though the Center will probably be unable to handle more than half-a-dozen workshops in any one summer, it hopes to spread across the country in the summer of 1962, offering workshops in the far west and the mountain states as well as the areas represented this year. Any educational institution wishing to sponsor such a workshop in the summer of 1962 is invited to communicate with the Center at the earliest opportunity. The Center is also ready to advise colleges and universities which wish to launch such workshops on their own, and will be glad to communicate the details of this summer's work, good and bad, to anyone who requests such information in the fall. A more general report on what happened in the workshops will appear in the fall issue of *Programed Instruction.*

The workshops will be collaborations between the Center and the host institution, and will therefore vary greatly from each other, according to the special facilities and personnel on the campus. Certain features, however, should be present in all of them. Basically the two-week workshops will be divided in half, the first part to be devoted to instruction in and discussion of programing, the second part to stress actual experience in the construction of programs.

Two books will figure in every workshop — *Teaching Machines and Programmed Learning,* by Lumsdaine and Glaser, which will be in the library, and *Learning: Reinforcement Theory,* by F. S. Keller, which is quite brief and which students will be expected to purchase. Members of the Center will be available for individual consultation on at least one afternoon during the first week.

Over the weekend and on the following Monday, each participant in the workshop will begin to program a unit in his own teaching area. On Tuesday, students will be made available to test the programs. From here until the final day, evenings will probably have to go to the revision of the first-draft programs. During the course of this week, the workshop will meet for one day as a unit to discuss, in the light of programing experiences, the relation of the "teaching machine" to programing. At least one other day of this week, and possibly two, will be spent with the students who are working over the participants' programs. During these sessions, staff members will be present in the background, ready to discuss problems with participants in the workshop.

On the last day of the workshop, the group will meet as a whole for discussion, review, and evaluation of the two weeks. It is possible that everyone will be a little tired by then; it is hoped that everyone will be more than a little excited.

FAULTY FRAMES — *(Continued from page 5)*

a) Is the frame badly written or ambiguous? Does it use words or phrases that are over the students' heads?

b) Has a faulty analysis of the subject matter led the student to answer incorrectly?

c) Have you previously taught a principle which is similar to this one, but which leads in this case to a wrong answer, and then failed to differentiate between the earlier sequence and the new sequence?

d) If the item is a review item, is it too far removed from the original sequence? Could you cure the difficulty by review items earlier in the sequence?

(Faulty Frames will be a regular feature of this bulletin. The Center will welcome contributions to this feature from all programers, full-time and part-time, who have been through such experiences. The contribution should include the mistake, the analysis, and the correction, with data on the number of students who tried both the faulty and the correct sequence, and the error rates on each.)